



微机系统与接口技术

主讲：许万茹



许万茹 博士，副教授，硕士生导师

- 2007年和2018年在北京交通大学计算机与信息技术学院分别获学士和博士学位。
- 2016-2017年美国伦斯勒理工大学从事研究工作。

授课：深度学习、汇编与接口技术、微机系统与接口技术

研究方向：计算机视觉、深度学习、图像/视频分析与理解、人体行为识别等。

在国内外期刊和会议上发表学术论文40余篇，包括

IEEE Trans. on Image Processing, IEEE Trans. on Multimedia, IEEE Trans. on Circuits and Systems for Video Technology, CVPR, ACM Multimedia 等。

主持国家自然科学基金（青年项目）、国家博士后基金面上项目、中央高校基本科研业务费等多个项目。

办公地点：土建楼424

Email: xwanru@bjtu.edu.cn

主页: <http://faculty.bjtu.edu.cn/9522/>



一、时间和地点

第二学期(2022-2023学年)																													
月份	二月		三月				四月				五月					六月				七月					八月				
学期	春季学期																		夏季学期										
周次	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	1	2	3	4	5	6	7	8	9	10	
星期一	20	27	6	13	20	27	3	10	17	24	1	8	15	22	29	5	12	19	26	3	10	17	24	31	7	14	21	28	
星期二	21	28	7	14	21	28	4	11	18	25	2	9	16	23	30	6	13	20	27	4	11	18	25	1	8	15	22	29	
星期三	22	1	8	15	22	29	5	12	19	26	3	10	17	24	31	7	14	21	28	5	12	19	26	2	9	16	23	30	
星期四	23	2	9	16	23	30	6	13	20	27	4	11	18	25	1	8	15	22	29	6	13	20	27	3	10	17	24	31	
星期五	24	3	10	17	24	31	7	14	21	28	5	12	19	26	2	9	16	23	30	7	14	21	28	4	11	18	25	1	
星期六	25	4	11	18	25	1	8	15	22	29	6	13	20	27	3	10	17	24	1	8	15	22	29	5	12	19	26	2	
星期日	26	5	12	19	26	2	9	16	23	30	7	14	21	28	4	11	18	25	2	9	16	23	30	6	13	20	27	3	

开学：2023年2月20日开始上课。

夏季学期（含暑假）：2023年6月26日—2023年9月1日

清明节：4月5日 劳动节：5月1日 端午节：6月22日

清明节、劳动节、端午节：届时结合国务院节假日安排和学校安排，以校办通知为准。



一、时间和地点

周二 SY205 (或 九教北402内 实验室) 第一节

周五 SY205 (或 九教北402内 实验室) 第一节

节次	星期一	星期二	星期三	星期四	星期五
第1节 08:00-09:50		M302016B [01] 微机系统与接口技术[本] 第01-16周 思源楼 SY205			M302016B [01] 微机系统与接口技术[本] 第01-16周 思源楼 SY205
第2节 10:10-12:00					
第3节 12:10-14:00					
第4节 14:10-16:00					
第5节 16:20-18:10					
第6节 19:00-20:50					
第7节 21:00-21:50					



北京交通大学
BEIJING JIAOTONG UNIVERSITY



二、联系方式



以及邮件、课程平台（作业）



北京交通大学
BEIJING JIAOTONG UNIVERSITY



三、做实验方式

线下实验： 汇编部分（自己的电脑） + 接口部分（**实验箱**）

地点：九教北402内 综合实验室



五、课程平台

1. 教务处的课程平台:

浏览作业、课件、MOOC视频, 提交作业、实验报告

2. 智慧教学课程 轻新课堂

学习“智慧教学课程平台
使用培训--学生版.ppt”



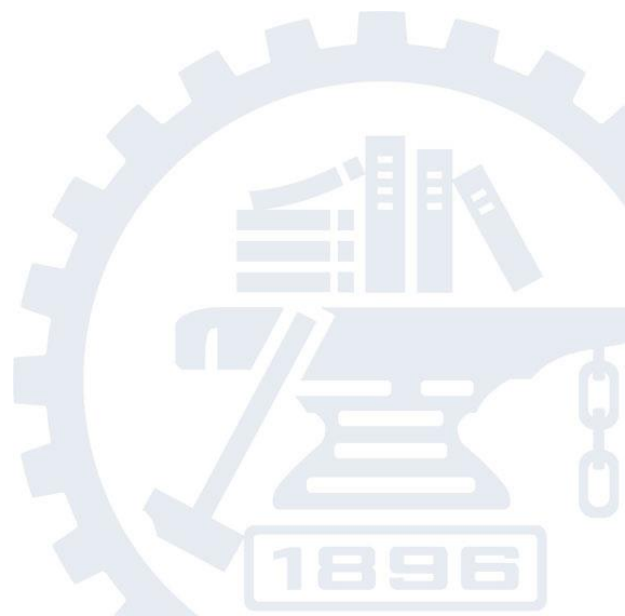


北京交通大学
BEIJING JIAOTONG UNIVERSITY



课程介绍

- 学习目标
- 学习内容
- 参考教材
- 课程安排
- 课程考核





北京交通大学
BEIJING JIAOTONG UNIVERSITY



课程特点

具有技术性、工程性和实践性等特点，是理论联系实际，提高实际应用能力的重要教学环节，它在培养高素质人才中具有重要的地位和作用。因此学生只有通过大量的实验才能取得良好的教学效果。



北京交通大学
BEIJING JIAOTONG UNIVERSITY



1、学习目标

软件是如何控制硬件的？

运用汇编程序对硬件接口进行控制





2、学习内容

- 了解微机系统内部运行机制
- 学会汇编语言编程的基本技术→工具本身
- 掌握接口的基本工作原理→问题的分析
- 掌握用汇编语言对接口的控制方法→如何用工具解决问题





北京交通大学
BEIJING JIAOTONG UNIVERSITY



3、课程教材

微机系统与接口技术 = 汇编语言 + 接口技术
主干课、必修课 必修课 必修课



《微机原理与接口技术》第三版

彭虎主编

电子工业出版社



参考教材

- ① 微机原理与接口技术(第3版), 彭虎等编著, 电子工业出版社。

参考资料:

- ① 微机原理与接口技术学习指导(第2版), 彭虎等编著, 电子工业出版社
- ② 微型计算机接口技术及应用, 刘乐善主编, 华中科技大学出版社, 2011年7月第1版。
- ③ 汇编语言程序设计 (第3版), 徐建民、邵艳华编著, 电子工业出版社, 2010年6月第3版。



课程面向的计算机类型

- 主要面向80X86机型，汇编语言涉及到8086的汇编语言，接口技术涉及到80X86的计算机，课程教学主要以16位机型为对象

(ARM等32位机型较为复杂，自学)

- 很多汇编语言实验可以在自己的X86电脑上完成





5、课程安排

- 理论教学**44学时**，实验教学**20学时**，总计**64学时**
- 理论教学中汇编约占**22学时**，接口约占**42学时**
- 实验教学包括常规实验（**12学时**，**6个实验**）和研究性实验（**8学时**，**1个实验**）
- 常规实验包括**2个**汇编课堂实验、**4个**接口课堂实验



6、课程考核

考试成绩占**50%** (开卷) , 平时成绩占**50%**

平时成绩包括:

■ 平时作业+考勤**11分**(4-5次), 要求:

●晚交没有分

• 作业 (word电子版) 提交到课程平台

作业名称格式为: 学号+姓名+第X章+作业

• 提交时间以网络提交时间为准



■ 常规实验**24分**

- 6个实验，每个实验4分，**实验检查2分+实验报告2分 (word)**
- 实验报告的提交要求同作业
- 实验报告电子版的名称格式为：学号+姓名+实验X+实验名称

■ 研究性实验**1个，15分**，实验报告的要求同常规实验





北京交通大学
BEIJING JIAOTONG UNIVERSITY



- **缺勤或作业不交共累计5次则取消考试资格。**
- **邮件/微信群答疑**
或者课前/课后答疑，其它答疑时间另行通知。





上机要求

- 上机前认真编写好实验程序，实验中要独立完成实验；
- 完成后，需要课上检查；（给分依据）
- 有问题鼓励自主探索，实在解决不了找老师答疑。

建 议

多读程序，多写程序，多上机调试程序并注意观测接口的状态，体会汇编对接口的控制作用。



第1章 基础知识

主讲：许万茹





主要内容

1.1 汇编语言

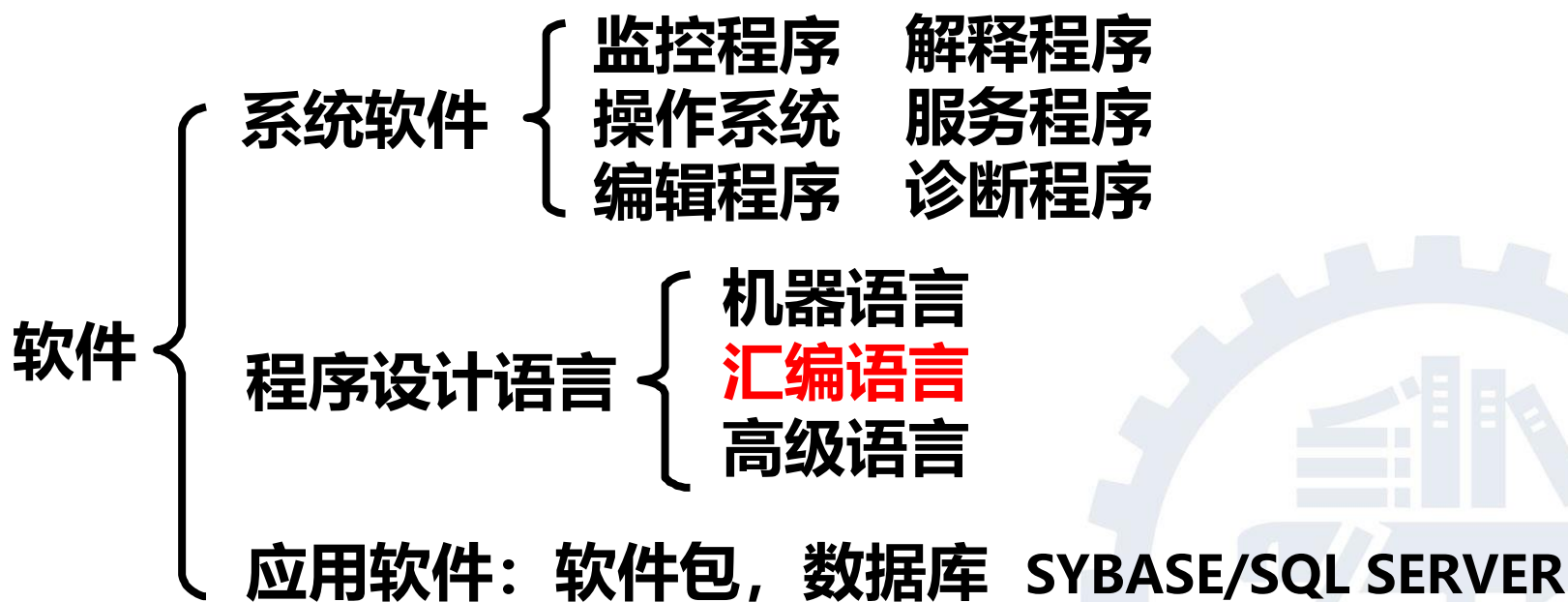
1.2 硬件接口

1.3 计算机基本知识



1.1 汇编语言

1、 计算机软件体系——汇编语言的地位





计算机语言的发展

机器语言

高级语言



FORTRAN

BASIC

python

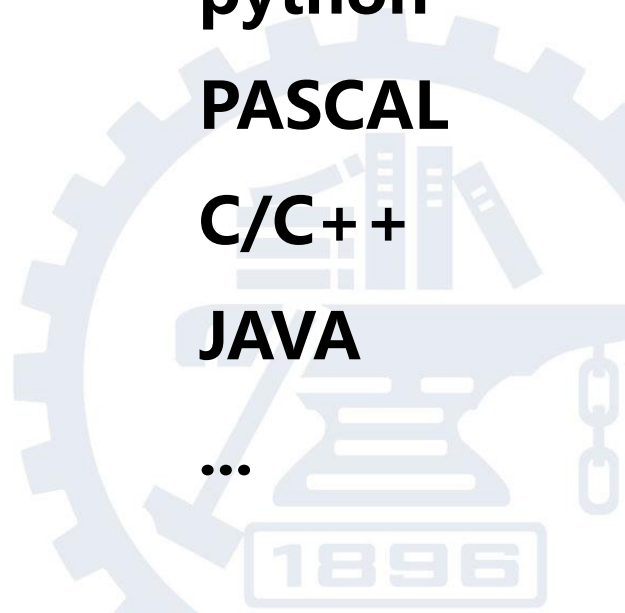
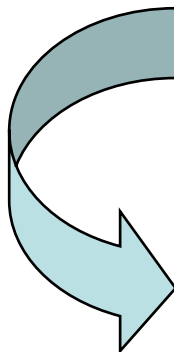
PASCAL

C/C++

JAVA

...

汇编语言





2、汇编语言的特点

- 面向机器的低级语言，具有专用性。
- 保持了机器语言的优点，具有直接和简捷的特点。
- 可有效地访问、控制计算机的各种硬件设备，如磁盘、存储器、CPU、I/O端口等。
- 目标代码简短，占用内存少，执行速度快，是高效的程序设计语言。
- 经常与高级语言配合使用，应用十分广泛。



3、汇编语言的应用

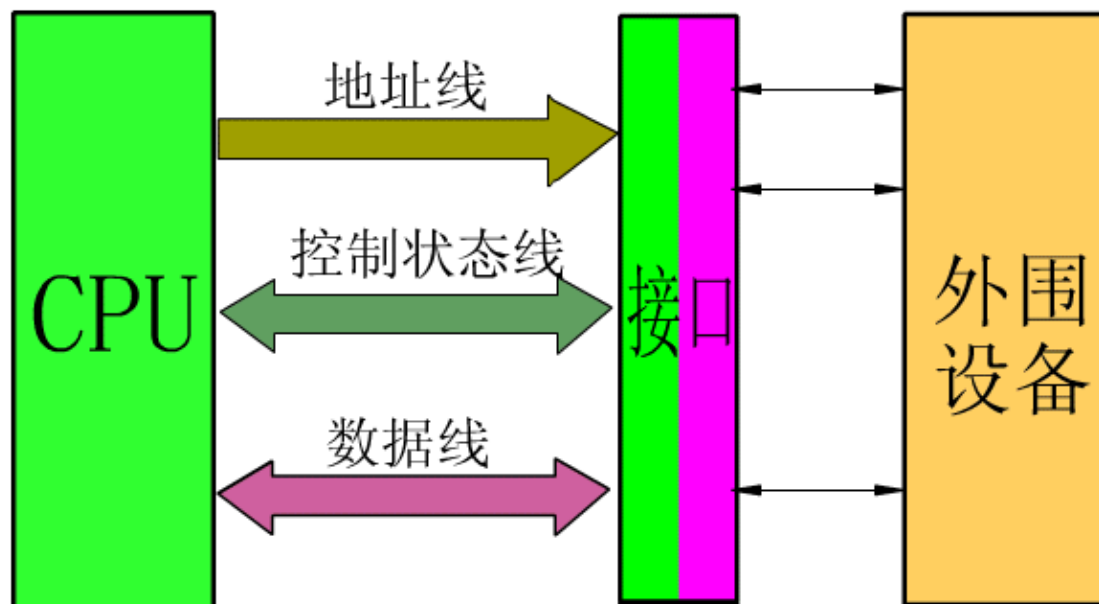
系统程序、高效率代码、I/O驱动程序

- 很多系统软件是用汇编语言编写的。
- 某些快速处理、位处理、访问硬件设备等高效程序是用汇编语言编写的。
- 高级绘图程序、视频游戏程序一般是用汇编语言编写的。

本课程的汇编以8086汇编语言为主



CPU、接口和外围设备间的连接



软件

信号转换设备

硬件控制信号





1、接口发展的几个阶段

- ❁ **无接口阶段**（微型企业：老板与员工）：**CPU与外围设备直接相连，直接控制；信号线不兼容，效率低，复杂，速度不兼容，耦合性强，不易扩展等缺点。**
- ❁ **简单接口阶段**（小型企业：老板，独立性很差的中层干部、员工）：**在CPU与外设间引入接口电路，可以使连接标准化；CPU（用户程序）通过接口对外设进行间接控制；**
- ❁ **复杂接口阶段**（大中型企业：老板，独立性较强的中层干部、员工）：**接口电路通过复杂的接口协议软件可较为独立地控制外围设备，控制过程对外设用户不可见/不可控（即接口的底层，如总线驱动程序），PCI/USB等就是复杂接口。用户对外设的控制需由用户程序（接口的用户层）和底层设备如PCI总线的驱动程序（接口的底层）共同配合完成。**



2、接口的功能

(1) 执行CPU命令

- 接口电路对CPU发来的命令信息（在命令寄存器中，称为**命令口**）进行识别和分析，分解成若干个控制信号，传送到I/O设备，使其产生具体的操作。
- CPU**不是**直接把命令送到被控对象，**而是**通过接口电路进行控制。

(2) 返回外设状态信息

接口执行CPU的命令后，将整个过程的状态存在接口的状态寄存器(**状态口**)中，供CPU读取。状态是：“忙”、“闲”、“就绪”、“错”等。



(3) 数据缓冲能力

避免因速度的差异而丢失数据。接口中一般都有数据缓冲寄存器，称为**数据口**。数据缓冲器分为输入和输出缓冲器两种。

(4) 信号转换功能

由于外设提供或需要的信号与CPU的总线不兼容，因此信号必须转换。

(5) 设备选择功能（大型的部门）

微机系统中一般有多个外设，一个外设中也有多个端口，接口中必须有端口选择能力。

(6) 数据宽度与数据格式转换的功能

如：并/串转换、串/并转换等。





3、接口的组成

(1) 接口逻辑电路

包括：命令寄存器，状态寄存器，数据缓冲寄存器
(数据收发速率不匹配时)。

可编程大规模集成芯片中都包含这些电路。

(2) 端口地址译码电路

作用是进行设备选择。需由用户自行设计。

(3) 附加电路

需根据接口不同的任务和功能添加的功能模块电路。



4、接口的软件编程

(1) 初始化程序段

对可编程接口芯片进行初始化，完成设置。

(2) 传送方式处理程序段

传送方式的处理。如：查询、中断、DMA等传送过程中的处理程序。

(3) 主控程序段

完成接口任务的程序段。

(4) 程序终止与退出程序段

包括程序结束前对接口中硬件的保护程序段。

(5) 配置段

包括人-机对话，菜单设计等内容。





5、CPU与接口交换数据的方式

- **查询方式：**包括无条件传送和程序查询传送。无条件传送不查询外设状态，认为外设已经准备就绪，直接与外设传送数据。程序查询传送在执行输入输出前，要先查询接口中状态寄存器的状态。
- **中断方式：**当外设作好传送准备后，主动向CPU请求中断，CPU响应中断后在中断处理程序中与外设交换数据。若外设未准备好，CPU可以执行其他程序，从而提高了CPU的利用率。
- **DMA方式：**DMA方式是一种由专门的硬件电路执行I/O的数据传送方式，它可以让外设接口直接与内存进行高速的数据传送，而不必经过CPU。这种专门的硬件电路称为DMA控制器，简称DMAC。



6、接口电路的分析与设计基本方法（如何当好中层干部？）

（1）两侧分析法

凡是接口都有两侧，**一侧是CPU**，**另一侧是外设**。

对CPU一侧就是**三总线**，因此分析比较容易，主要搞清楚是什么类型的CPU以及数据总线、地址总线的宽度和控制线的逻辑定义、时序关系。

对外设一侧情况很复杂，这是因为被控对象外设种类繁多，所提供的信号线五花八门；其逻辑定义、时序关系、电平高低差异甚大。要**搞清两点**：**一是外设的信号引脚的功能定义和逻辑定义（外部表现）**，**二是了解被控外设的工作过程（内部原理）**。



(2) 硬软结合法

① 硬件设计方法（接口硬件有一定设计自由度）

- ◆ 合理选用外围接口芯片。
- ◆ 如有需要，需在接口芯片之外设计附加电路。

② 软件设计方法（接口硬件固定）

- ◆ 采用汇编语言（或高级语言）直接对底层硬件编程，多用于控制非微机标准设备。
- ◆ 采用DOS功能调用和BIOS调用编程，多用于控制微机标准设备。



北京交通大学
BEIJING JIAOTONG UNIVERSITY



1.3 计算机基本知识

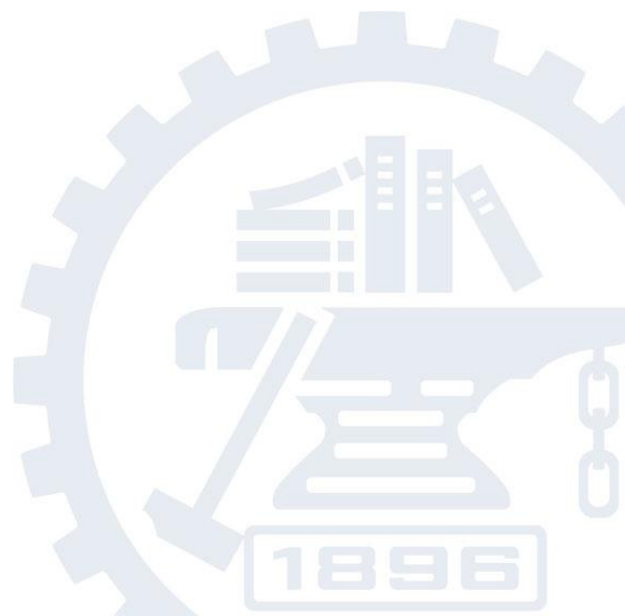
1.3.1 微型计算机系统的组成

1.3.2 微型计算机总线

1.3.3 计算机中信息表示方法

1.3.4 进制及其转换

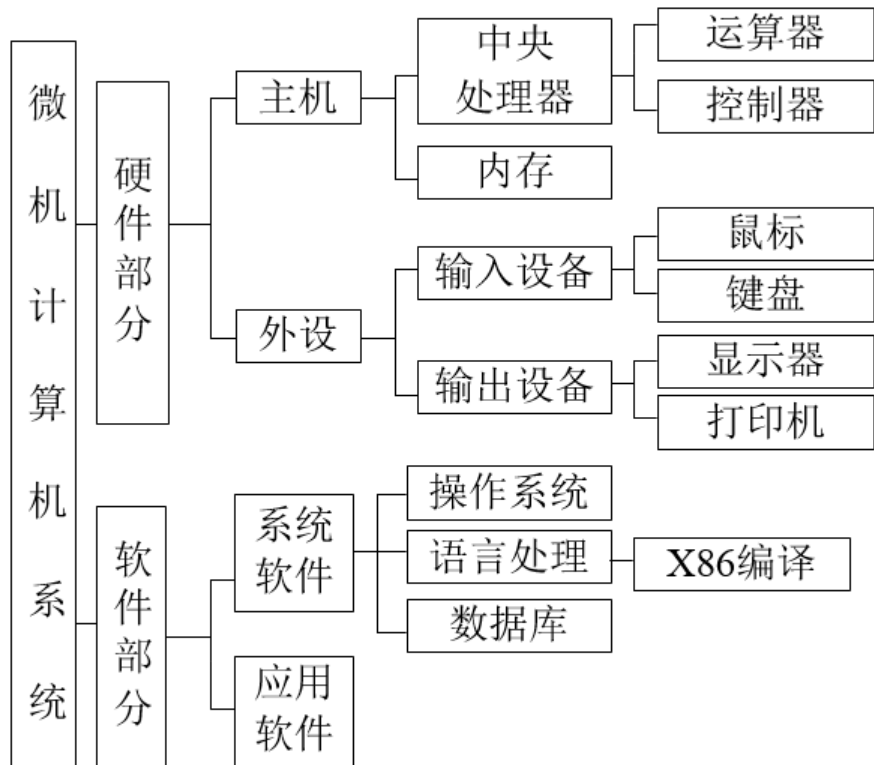
1.3.5 二进制运算规则





1.微型计算机系统的组成

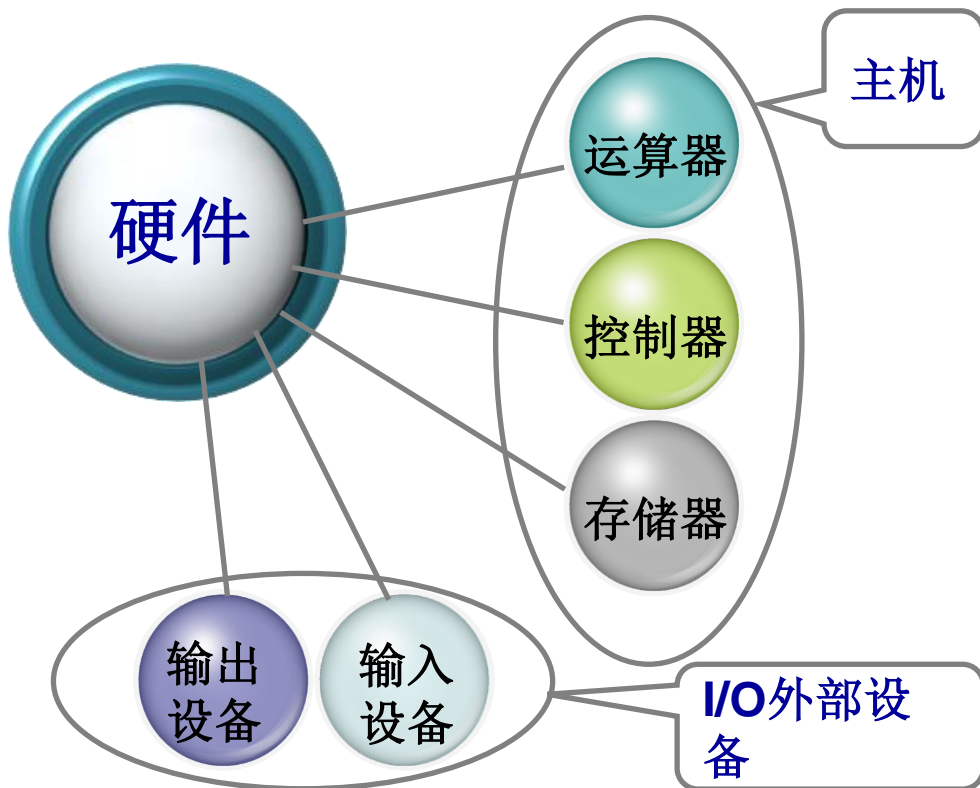
微型计算机系统由硬件部分和软件部分组成



- 硬件部分主要由主机和外部设备组成
 - ✓ 主机：中央处理器、内存
 - ✓ 外部设备：输入和输出
- 中央处理器包括运算器、控制器
- 内存用于存储程序、数据和文件
- 输入设备：将信息从外部输入到计算机内
- 输出设备：将信息从计算机内输出到外部
- 软件分为系统软件和应用软件
 - ✓ 系统软件：控制和协调计算机及外部设备, 支持应用软件开发和运行
 - ✓ 应用软件：用户可以使用的各种程序设计语言，以及用各种程序设计语言编制的应用程序的集合



1. 微型计算机系统的组成



5个组成部件

- 运算器: 算术运算和逻辑运算
- 控制器: 对程序控制信息分析控制并协调输入, 输出操作或内存访问
- 存储器 (内存): 实现记忆功能
- 输入设备: 原始数据的输入
- 输出设备: 计算结果输出





软件系统

操作系统

数据库

机器指令

CPU可以识别的、用于控制**CPU**完成特定操作的二进制编码的指令。

汇编语言

机器指令的助记符表示-本课程是**X86**汇编语言

高级语言

人为设计的、更接近人类自然语言描述习惯的计算机语言，如**VB**、**C/C++**。

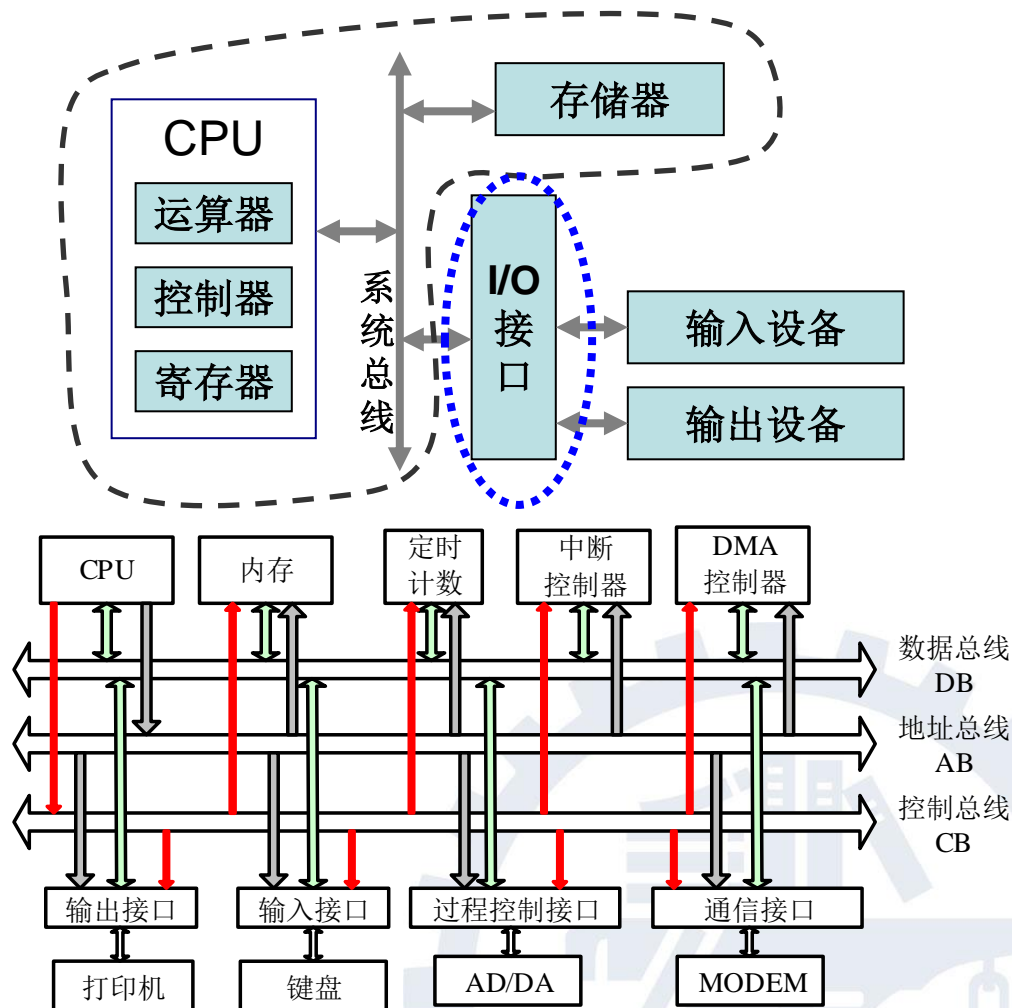
语言处理

MASM
VB
C/C++
JAVA



2 微型计算机系统总线

- 微型计算机是以总线结构来连接各个功能部件的。
- 总线是计算机各种功能模块传送数据信息的公共通道
- 总线用来传输数据信息、地址信息和控制信号的，分别称为数据总线DB，地址总线AB和控制总线CB。
- 外部设备通过相应的接口电路与扩展总线相连接-ISA总线





3 计算机中信息的表示方法

进制基本概念

- 进制也就是进位计数制，比如我们人类日常使用的是十进制数，就是逢10进1；计算机中的数，是以二进制表示，逢2进1
- 进制的表示方法是：由一组数码符号与位权完成
 - ✓ 数码符号与进制有关，十进制有10个数码，即0-9；二进制有二个数码0和1；十六进制有16个数码0-9，和A-F
 - ✓ 位权：进位制中每位对应的单位值-权值

具体二进制的表示方法：

- 记数按逢二进一
- 有两个数码，0和1
- 二进制中第k位的权值是 2^k 。

二进制： $b_7 \ b_6 \ b_5 \ b_4 \ b_3 \ b_2 \ b_1 \ b_0$

例如： 1 0 1 0 0 1 0 1

$2^7 \ 2^6 \ 2^5 \ 2^4 \ 2^3 \ 2^2 \ 2^1 \ 2^0$

权值： 128 64 32 16 8 4 2 1

十进制： $d_3 \ d_2 \ d_1 \ d_0$

例如： 6 0 8 0 D

$10^3 \ 10^2 \ 10^1 \ 10^0$

权值： 1000 100 10 1



3 计算机中信息的表示方法

有符号数和无符号数

- 有符号数和无符号数是针对二进制来讲的
- 有符号数就是用最高位表示符号：最高位是0，表示正数；1是负数；其余数位是数值；无符号数全部二进制均代表数值，没有符号位：例如1011B有符号数是-3D，无符号数是11D，而0011B有符号数和无符号数都表示+3D
- 无符号数的加减运算结果以进位位CF表示是否超出字长范围，产生进位；
- 有符号数的加减运算结果以溢出位OF表示是否超出字长范围，产生溢出；
- 对于8位数据：有符号数表示范围是-128~127；无符号数表示范围是0~255

十六进制	转换为二进制									无符号数加	有符号数加
7H		0	0	0	0	0	1	1	1	7	7
FBH		1	1	1	1	1	0	1	1	251	-5
102H	1	0	0	0	0	0	0	1	0	258	2
										CF=1进位	OF=0无溢出

十六进制	转换为二进制									无符号数加	有符号数加
7H		0	0	0	0	0	1	1	1	7	7
7CH		0	1	1	1	1	1	0	0	124	124
83H	0	1	0	0	0	0	0	1	1	131	131
										CF=0无进位	OF=1溢出

在编程之前就要清楚是对有符号数还是无符号数进行计算操作，以便程序采用不同指令，即依据进位还是溢出进行判断



3 计算机中信息的表示方法

ASCII码

- ASCII码表示美国信息交换标准码：即：用一个字节来表示一个字符，低7位是字符的ASCII值，最高位一般用作校验位。‘A’:41H, ‘a’:61H; ‘1’:31H
- 如果显示的数字是0~9，则将此数加上30H，转换成相应的ASCII码30H~39H输出即可显示数字0~9。比如要显示数字1，则需要输出ASCII码值是31H;
- 如果是A~F，则应将此数加上37H，转换成ASCII码41H~46H输出即可显示A~F。比如要显示F，则需要输出ASCII码值是FH+37H=46H，即输出ASCII码值46H，则在显示器中显示大写的F;
- 如果是a~f，则应加上57H，转换成ASCII码61H~66H输出即可显示a~f。



4 进制及其转换

数值进制转换

- 2进制、16进制、十进制之间的转换
- 二进制后缀B，十进制后缀D，十六进制后缀H。
- 一位十六进制与二进制、十进制之间的转换表

0AH=10D=1010B

非数转换-ASCII码

美国信息交换标准代码，使用7位二进制对字母、数字、常用符号和控制字符进行编码。

其中：00H~1FH为控制字符；30H~39H为数字

41H~5AH为大写字母；61H~7AH为小写字母

其余为常用符号。

二进制	十六进制	十进制
0000	0	0
0001	1	1
0010	2	2
0011	3	3
0100	4	4
0101	5	5
0110	6	6
0111	7	7
1000	8	8
1001	9	9
1010	A	10
1011	B	11
1100	C	12
1101	D	13
1110	E	14
1111	F	15



4 进制及其转换

二进制、十六进制之间的转换

按从低到高每4位二进制对应一位十六进制转换。

$$\underline{100} \ \underline{1010} \ \underline{0001} \ \underline{0011} \text{ B} = 4\text{A}13 \text{ H}$$

$$14\text{BF H} = \underline{0001} \ \underline{0100} \ \underline{1011} \ \underline{1111} \text{ B}$$

二进制、十六进制转换为十进制

按数码值乘以权值，再相加的方法转换

$$1010 \text{ B} = 1 \times 8 + 0 \times 4 + 1 \times 2 + 0 \times 1 = 10 \text{ D}$$

二进制权值： 8 4 2 1



4 进制及其转换

十进制转换成二进制、十六进制

使用连除，将十进制数连续除2或16，直到商为0。连除所得的余数的反向排列即转换的结果。

➤ 自称为“权值匹配法”

✓ 方法：将十进制表示为二进制权值的和，在和中参与相加的权值对应位为1（对十进制数有“贡献”的位），不参加相加的位为0（无“贡献”的位）。

✓ 例如：将10D转换为二进制数。

10D表示为权值的和，就是8+2：

权值	8	4	2	1
----	---	---	---	---

二进制数	1	0	1	0
------	---	---	---	---

即：10D=1010B



5 运算规则-二进制的算数运算

二进制运算规则-算数运算

- 计算机采用二进制计算形式。
- 有两种二进制运算，即算术运算和逻辑运算。
- 二进制数的基本算术运算包括加、减、乘、除四则运算
- 二进制数的加法：根据“逢二进一”规则进行运算
- 二进制数的减法：根据“借一有二”的规则进行运算



加	减	乘	除
$0+0=0$	$0-0=0$	$0\times 0=0$	$0\div 0$ 非法
$0+1=1$	$0-1=1$ 借 1	$0\times 1=0$	$0\div 1=0$
$1+0=1$	$1-0=1$	$1\times 0=0$	$1\div 0$ 非法
$1+1=0$ 进 1	$1-1=0$	$1\times 1=1$	$1\div 1=1$



5 运算规则-二进制的逻辑运算

二进制运算规则-逻辑运算

- 逻辑运算通常用来测试真假值。正逻辑：1为真，0为假
- 逻辑运算包括与、或、非等基本逻辑运算
- 运算法则：
 - ✓ 与运算：同为真时为真
 - ✓ 或：同为假时为假
 - ✓ 异或：相同为假

逻辑
运算

与	或	非	异或
$0 \wedge 0 = 0$	$0 \vee 0 = 0$	$\overline{0} = 1$	$0 \oplus 0 = 0$
$0 \wedge 1 = 0$	$0 \vee 1 = 1$		$0 \oplus 1 = 1$
$1 \wedge 0 = 0$	$1 \vee 0 = 1$	$\overline{1} = 0$	$1 \oplus 0 = 1$
$1 \wedge 1 = 1$	$1 \vee 1 = 1$		$1 \oplus 1 = 0$



5 运算规则-有符号数和无符号数

运算规则-有符号运算

- 对于8位的有符号数，其取值范围在-128~127之间
 - 有符号数的运算超出取值范围-溢出
- 例：两个有符号数7和124相加，相加之后的值是131，而有符号数表示最大值是127，超出了有符号数表示范围，所以是溢出，即OF=1

运算规则-无符号运算

- 对于8位的无符号数，其取值范围在0~255之间
 - 无符号数的运算超出取值范围-进位
- 例：两个无符号数7和251相加，相加之后的值应该是258，而无符号数只能表示的最大值是255，超出了无符号数的表示范围所以是进位，即CF=1